Noname manuscript No.
(will be inserted by the editor)

Unsupervised Person Re-identification: Clustering and Fine-tuning

Hehe Fan · Liang Zheng · Yi Yang

Received: date / Accepted: date

Abstract The superiority of deeply learned pedestrian representations has been reported in very recent literature of person re-identification (re-ID). In this paper, we consider the more pragmatic issue of learning a deep feature with only a few or no labels. We propose a progressive unsupervised learning (PUL) method to transfer pretrained deep representations to unseen domains. Our method is easy to implement and can be viewed as an effective baseline for unsupervised feature learning. Specifically, PUL iterates between 1) pedestrian clustering and 2) fine-tuning of the convolutional neural network (CNN) to improve the original model trained on the irrelevant labeled dataset. At the beginning when the model is weak, CNN is fine-tuned on a small amount of reliable examples which locate near to cluster centroids in the feature space. As the model becomes stronger in subsequent iterations, more images are being selected as CNN training samples. Progressively, pedestrian clustering and the CNN model are improved simultaneously until algorithm convergence. We then point out promising directions that may lead to further improvement.

Keywords Person re-identification · Unsupervised learning · Deep learning

1 Introduction

This paper discusses person re-identification (re-ID), a task which provides critical insights whether a person re-appears in another camera after being spotted in one [37]. Recent works view it as a search problem aiming to retrieving the relevant images to the top ranks [35,38].

Large-scale learning methods based on the convolutional neural network (CNN) constitute the recent advances in this field. Broadly speaking, current deep learning methods can be categorized into two classes, *i.e.,* similarity learning and representation learning. For the former, the training input can be image pairs [25], triplets [18], or quadruplets [4]. These methods directly output the similarity score of two input images by focusing on their distinct body patches, without an explicit feature extraction process. This line of methods are less efficient during testing. On the other hand, when learning feature representations, contrastive loss or softmax loss are most commonly employed. For these methods, there is an explicit feature extraction process for the query and gallery images, which is learned to discriminate among a pool of identities or among image pairs/triplets [5,11]. Then the re-ID procedure is rather like a retrieval problem under some commonly used similarity measurement such as Euclidean distance. Due to its efficiency in large-scale galleries and the potentials in further acceleration, the deep feature learning strategy has become more popular [38,37,11,29]. Therefore, we adopt the classification model proposed in [37,29] for feature learning.

Despite the impressive progress achieved by the deep learning methods [38,16,11], most of these methods focus on re-ID with sufficient training data in the same environment. In fact, the lack of labeled training data under a new environment has been addressed by many previous works by resorting to unsupervised learning [13,20,19,31]. These works typically deal with the relatively small datasets [9,28] and to our knowledge, none of these works are integrated into the deep learning framework that has been proven as the state of the art under the large-scale datasets [35,11,24,8]. In fact, the problem of high annotation cost is more severe for deep learning based methods due to the intensive demand of data. So under this circumstances, we make initial attempts by developing a easy-to-implement method for unsupervised deep representation learning in the re-ID community. Our method can be easily extended to the semi-

Hehe Fan
University of Technology Sydney

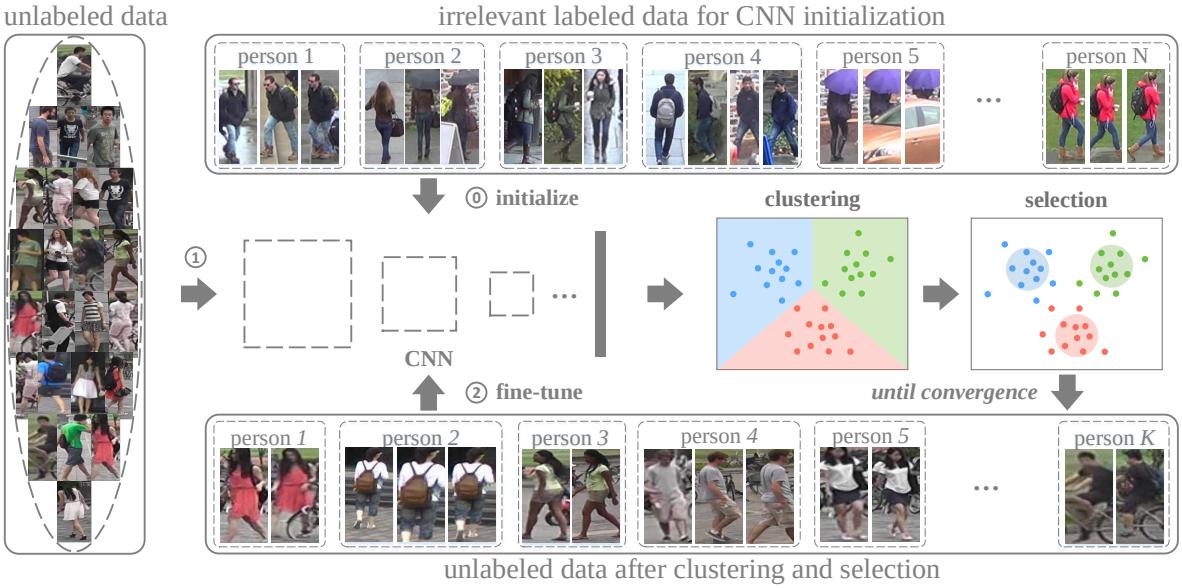


Fig. 1 Illustration of the PUL framework. In step 0, we initialize CNN on an irrelevant labeled dataset. Then we go into the iterations. During each iteration, we 1) extract CNN features for the unlabeled dataset, and perform clustering and sample selection, and 2) fine-tune the CNN model using the selected samples. Each cluster denotes a person.

supervised setting by annotating a small amount of bounding boxes. We also notice some works on human-in-the-loop learning [26, 17], introducing labels by human feedback. We envision our work is complementary to this line of methods.

In this paper, we aim at finding "better" feature representations without labeled training data to adapt to the unlabeled dataset. Named "progressive unsupervised learning" (PUL), our method is an iterative process and each iteration consists of two components (shown in Fig. 1). 1) We deploy the model pre-trained on some external labeled data to extract features of images from the unlabeled training set. Standard k-means clustering is performed on the CNN features, followed by a selection operation to choose reliable samples. 2) The original model is fine-tuned using the selected training samples. We then use this newborn model to extract features and begin another iteration of training. At the beginning, when the model is weak, the proposed approach learns from a small amount of reliable examples, which are near to cluster centroids in the feature space, to avoid getting stuck at a bad optimum. As the model becomes stronger in subsequent iterations, more data instances are adaptively selected and exploited as training examples. By this progressive method, the model and clustering are trained and rectified alternately. This process can also be called self-paced learning [14]. Among the early efforts, we propose an easy-to-implement unsupervised deep learning framework for large-scale person re-ID.

2 Related Works

Deeply learned person representations. Deeply learned person representations are producing state-of-the-art re-ID performance recently. A survey of re-ID can be viewed in [37]. Comparing with deep similarity learning methods [25, 18, 4], the representation learning methods are more extendable for large galleries. For example, Hermans *et al.* [11] used an efficient variant of the triplet loss based on the distance between samples. Another popular choice consists in training an identification network and extracting the intermediate output as a discriminative embedding [37, 29, 33, 34]. For example, Xiao *et al.* [30] propose an online instance matching loss to address the problem of having only few training samples each class. Lin *et al.* [16] combine attribute classification losses and the re-ID loss for embedding learning. In order to exploit more training data, Zheng *et al.* [38] employ the generative adversarial network to generate samples which are expected to generate uniform prediction probabilities in the softmax layer. In this paper, we adopt the baseline identification model, named "ID-discriminative Embedding" (IDE) in [33].

Unsupervised person re-ID. Although deeply learned person representations have achieved significant progress by supervised learning methods, less attention is paid on unsupervised learning for re-ID. Kodirov *et al.* [13] utilize a graph regularized dictionary learning to capture discriminative cues for cross-view identity matching. Yang *et al.* [31] propose a weighted linear coding method to learn multi-level descriptors from raw pixel data in an unsupervised man-

ner. Wang *et al.* [27] use a kernel subspace learning model to learn cross-view identity-specific information from unlabeled data. Peng *et al.* [20] propose a multi-task dictionary learning model to transfer a view-invariant representation from a number of existing labeled source datasets to an unlabeled target dataset. Ma *et al.* [19] utilize image-sequence information to fuse different descriptions. These methods focus on small-scale datasets and do not adopt deep features.

Self-paced learning. In the human learning process, knowledge is imparted in the form of curriculum and organized from easy to difficult. Inspired by this process, the theory of Curriculum Learning (CL) [3] is proposed to accelerate the speed of convergence of the training process to a minimum. The main challenge of CL is that it requires a known separation to indicate whether a sample in a given training dataset is easy or hard. To alleviate this deficiency, Self-Paced Learning (SPL) [14,12,6] embeds easiness identification into model learning. The learning principle behind SPL is to prevent latent variable models from getting stuck in a bad local optimum. Then SPL is widely used in weakly-supervised and semi-supervised learning. In this paper, we utilize SPL to select reliable samples to fine tune the original model. If we consider the labels of the samples from the unlabeled dataset as latent variables, our method also belongs to latent variable model. Therefore, it is necessary to avoid getting stuck in bad optimums using SPL.

3 Progressive Unsupervised Learning

In this section, we present our progressive unsupervised learning framework for re-ID in detail. The proposed approach begins with a basic convolutional neural network, such as the ResNet-50 [10] network trained on the *ImageNet*. Firstly, we fine tune the basic model by an irrelevant labeled dataset, which is stored as the original model. This step is also called original model initialization. Secondly, the original model is used to extract feature for the samples from the unlabeled dataset. The data instances are clustered and selected to generate a reliable training set. Thirdly, the original model is fine tuned by this reliable training set. Lastly, we use the new model to extract feature, and the second and third steps are repeated until the scale of the reliable training set becomes stable. The progressive unsupervised learning framework is demonstrated in Fig 1.

3.1 Formulation

Suppose the unlabeled dataset contains N cropped person images $\{x_i\}_{i=1}^N$ with K identities. Since the data is not labeled, we consider the identities $\{y_i\}_{i=1}^N$ of $\{x_i\}_{i=1}^N$ as latent variables.

Let v_i be the selection indicator of sample x_i. If v_i equals to 1, x_i is selected as a reliable sample; otherwise, x_i is discarded when fine tuning the original model. We further denote $\mathbf{v} = [v_1, \dots, v_N]$ as the selection indication vector and $\mathbf{y} = [y_1, \dots, y_N] \in \{1, \dots, K\}^N$ as the label vector for $\{x_i\}_{i=1}^N$. We formulate our idea as optimizing the following three problems alternately:

$$\min_{\mathbf{y}, \mathbf{c}_1, \dots, \mathbf{c}_K} \sum_{k=1}^{K} \sum_{y_i=k} \|\phi(x_i; \boldsymbol{\theta}) - \mathbf{c}_k\|^2. \tag{1}$$

$$\begin{aligned} \min_{\mathbf{v}} & \sum_{k=1}^{K} \sum_{y_i=k} v_i \|\phi(x_i; \boldsymbol{\theta}) - \mathbf{c}_k\| - \lambda \|\mathbf{v}\|_1, \\ & s.t.\ v_i \in \{0, 1\}; \sum_{y_i=k} v_i \geq 1, \forall k. \end{aligned} \tag{2}$$

$$\min_{\boldsymbol{\theta}, \mathbf{w}} \sum_{i=1}^{N} v_i \mathcal{L}(y_i, \phi(x_i; \boldsymbol{\theta}); \mathbf{w}), \tag{3}$$

where $\mathbf{c}_k$ is the mean of points $\{\phi(x_i; \boldsymbol{\theta})\}_{y_i=k}$ in the feature space, λ is a positive parameter and $\mathcal{L}$ denotes a loss function for classification.

3.2 Optimization Procedure

The optimization Eq. 1, Eq. 2 and Eq. 3 can be solved in an alternate manner.

1) Optimize $\mathbf{y}$ *and* $\mathbf{c}_1, ..., \mathbf{c}_K$ *when* θ *and* $\mathbf{v}$ *are fixed.* In this step, the optimization problem reduces to the classic k-means clustering and can be easily solved.

2) Optimize $\mathbf{v}$ *when* $\boldsymbol{\theta}$*,* $\mathbf{y}$ *and* $\mathbf{c}_1, ..., \mathbf{c}_K$ *are fixed.* The goal of this step is to select reliable data instances to fine tune the CNN model with the clustering results $\mathbf{y}$ in Eq. 1. A sample will be selected if the distance of its feature to its corresponding cluster centroid is smaller than λ. Here, λ is a threshold to control the reliable sample selection. The constraint $\sum_{y_i=k} v_i \geq 1$ can guarantee that each cluster contains at least one reliable sample.

3) Optimize $\boldsymbol{\theta}$ *and* $\mathbf{w}$ *when* $\mathbf{v}$*,* $\mathbf{y}$ *and* $\mathbf{c}_1, ..., \mathbf{c}_K$ *are fixed.* This step is to fine tune the model by the clustering and selection results from Eq. 1 and Eq. 2. Generally speaking, there are three types of loss functions for CNN models are commonly employed in the community. The first one is the classification method, in which a classifier is added at the end of model $\phi(x; \boldsymbol{\theta})$ as a fully-connected layer. When v_i equals to 0, the loss will not be calculated, which means the corresponding sample will not be used to fine tune the original model. Eq. 3 can be replaced with the contrastive loss [21] or triplet loss [23]. In this paper, we mainly focus on the classification model.

Fig. 2 Two visual examples of the working mechanism of progressive unsupervised learning (PUL). Each color denotes a distinct ID and the red circles represent reliable areas. From training iteration 2 to training iteration 3, more samples are selected reliable samples. In PUL, when the number of selected samples is saturated, the algorithm reaches convergence.

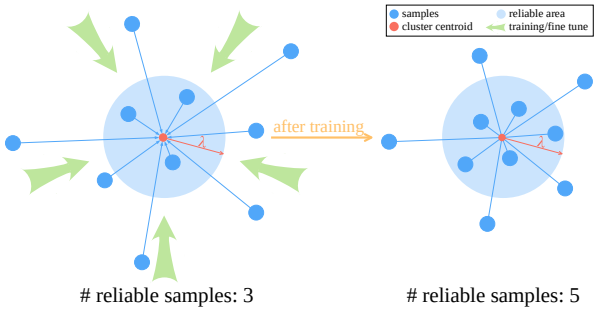


Fig. 3 Illustration of the baseline learning process. Reliable samples covered in the large blue circle are used for CNN fine-tuning. As the CNN model gets fine-tuned, more challenging training data can be mined.

Discussions. Eq. 1, Eq. 2 and Eq. 3 constitute a training iteration. As training goes on, we obtain more discriminative CNN models which facilitate the clustering process with smaller clustering errors. That is, with the optimization of Eq. 1 and Eq. 3, the clustering error $\|\phi(x_i;\boldsymbol{\theta})-\mathbf{c}_k\|^2$ will become smaller and smaller. This leads to an "self-paced" scheme in which more selection indicators v_i are expected to become 1 in order to minimize 2. Therefore, more and more examples are selected into the training set, until no more reliable data instances can be added. Fig. 3 illustrates how the training set changes during optimization.

In practice, we use cosine to measure the similarity between two feature vectors. The optimization algorithm of PUL is presented in Alg. 1. To guarantee each cluster contains at least one reliable sample, we choose the nearest feature to the corresponding centroid as the center of the cluster. In the algorithm, samples with $\mathbf{f}_i \cdot \mathbf{f}_k > \lambda$ will be selected for fine-tuning ($v_i = 1$); otherwise, sample will not be selected ($v_i = 0$). When the number of selected reliable samples is saturated, the algorithm will converge.

Algorithm 1: Progressive Unsupervised Learning

Input : Unlabeled data $\{x_i\}_{i=1}^N$;
Reliability threshold λ;
Number of clusters K;
Original model $\phi(\cdot,\boldsymbol{\theta}_o)$.
Output: Model $\phi(\cdot,\boldsymbol{\theta}_t)$.
Initialization: $\boldsymbol{\theta}_o \rightarrow \boldsymbol{\theta}_t$.

```
while not convergence do
    randomly initialize w;
    extract feature: f_i = φ(x_i, θ_t) for all i;
    k-means: update {y_i}_{i=1}^N and {c_k}_{k=1}^K;
    select center features: {c_k}_{k=1}^K → {f_k}_{k=1}^K;
    l2-normalize {f_i}_{i=1}^N;
    for k = 1 to K do
        for i = 1 to N do
            calculate inner product δ_i = f_i · f_k;
            if δ_i > λ then
                v_i ← 1; // selected
            else
                v_i ← 0; // removed
            end
        end
    end
    fine-tune < φ(·, θ_o), w > with selected samples → θ_t;
end
```

4 Experiments

4.1 Datasets and Settings

We mainly evaluate the proposed method on DukeMTMC-reID [38] and Market-1501 [35], because they are two relatively large datasets with multiple cameras. We also report results on CUHK03 [15], as complementary comparisons with other unsupervised learning methods. We do not use MARS [33], another large-scale dataset, because it is an extension of Market-1501 and hence has a similar data distribution with Market-1501.

DukeMTMC-reID is a subset of the pedestrian tracking dataset DukeMTMC [22]. This dataset contains 36,411 images with 1,812 identities captured from 8 different view points. Following Market-1501, the dataset is also split into three parts: 16,522 images with 702 identities for training, 17,661 images with 1,110 identities in gallery, and another 2,228 images with 702 identities in the gallery for query. Images in this dataset vary in size. For simplicity, we use "Duke-reID" to represent this dataset.

Market-1501 contains 32,668 images of 1,501 identities in total, which are captured from 6 cameras placed in front of a campus supermarket. The images are detected and cropped using the Deformable Part Model (DPM) [7]. The dataset is split into three parts: 12,936 images with 751 identities for training, 19,732 images with 750 identities for gallery, and another 3,368 hand-drawn images with the same 750 gallery identities for query. All the images are of the size 128×64.

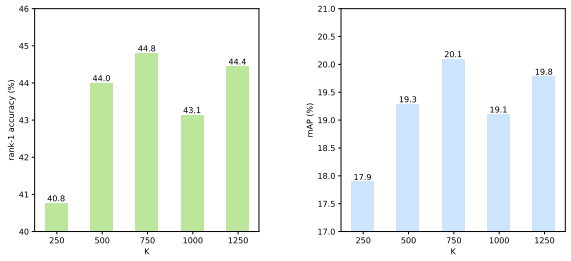
CUHK03 contains 14,096 images of 1,467 identities. Each identity is captured from 2 cameras in the CUHK campus, and has an average of 4.8 images in each camera. The dataset provides both manually labeled images and DPM-detected images. We experiment on the DPM-detected images. Note that the original evaluation protocol of CUHK03 has 20 train/test splits. Nevertheless, to be in consistency with Market-1501 and Duke-reID, we use the train/test protocol proposed in [39]: 7,365 images with 767 identities for training, 5,332 images with the remaining 700 identities for gallery, and 1,400 images with the same 700 gallery identities for query. Images in this dataset also vary in size.

We report the rank-1 accuracy and mean average precision (mAP) for all the three datasets.

4.2 Implementation Details

We use ResNet-50 [10] pre-trained on *ImageNet* as our basic CNN model. To fine tune the model using the classification framework [37], we modify the fully-connected layer to adapt to different datasets. We also insert a dropout layer before the fully-connected layer and set the dropout rate to 0.5 for all datasets. All images are resized to 224×224. For data augmentation, we randomly rotate images within 20 degrees and shift them horizontally and vertically within 45 pixels. Batch size is set to 16. We use stochastic gradient descent with a momentum of 0.9 and the learning rate is set to 0.001 without any decay during the whole training process. Unless otherwise specified, ResNet-50 is fine-tuned for 20 epochs within each PUL iteration.

During feature extraction, for a given image, we use the output of the average-pooling layer as the visual representation. The l_2 normalized representations are used in sample selection and retrieval, while clustering uses features without normalization (l_2 normalization in clustering yields inferior results in our preliminary experiments).



(a) (b)

Fig. 4 Impact of the number of clusters K on re-ID accuracy. ($\lambda = 0.85$). (a): rank-1 accuracy. (b): mAP.

For the clustering step, we use k-means++ [1] to select initial cluster centers for k-mean clustering to speed up convergence. The maximum number of iterations of the k-means algorithm within each PUL iteration is set to 300.

4.3 Performance of the Baseline System

The basic ResNet-50 model is fine-tuned for 40 epochs on each training set. Note that our focus is not on how to improve the performance by modifying neural networks [38, 24] or post-processing [39, 2, 32, 36]. Therefore, results in Table 1 do not represent the state of the art.

The major observation from Table 1 is that when the fine-tuned model is directly deployed on other datasets, re-ID accuracy is far inferior to training/testing on the same dataset. For example, the CNN model fine-tuned and tested on Market achieves 76.2% in rank-1 accuracy, but drops to 21.9% when trained on Duke and tested on Market. This is because of the changes in data distribution between different datasets.

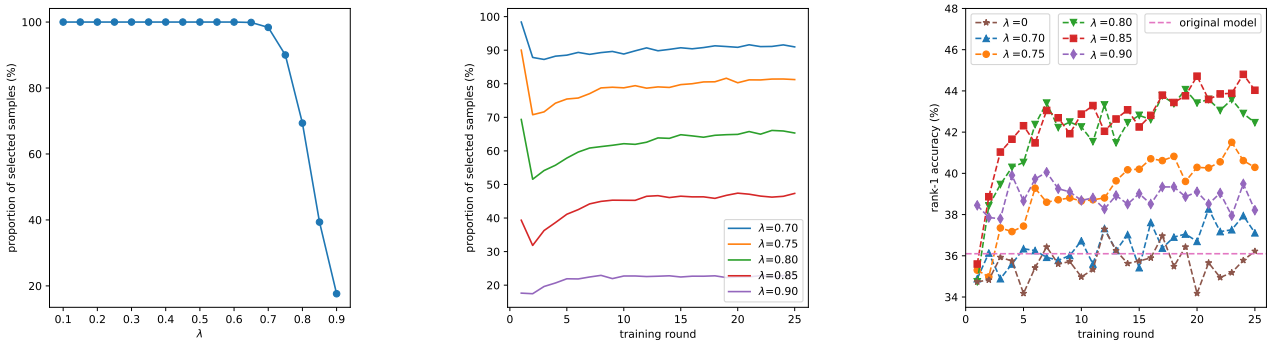
4.4 Evaluation of PUL

PUL vs. baseline We first compare PUL with deploying the fine-tuned CNN model on an unseen testing dataset (baseline). Results are shown in Table 1. PUL effectively improves the re-ID accuracy over the baseline. For example, when using Duke to train the original CNN model, PUL leads to an improvement of +8.6% in rank-1 accuracy and +5.9% in mAP on Market-1501.

Ablation study: PUL without sample selection. The selection of reliable samples for CNN fine-tuning is a key component in the PUL system. Here we investigate how the system works without reliable sample selection. In our experiment, for each PUL iteration, we use all the samples as training data for CNN fine-tuning. That is, after k-means clustering, all the images in each cluster are viewed as a class and fed into CNN.

features	Duke-reID					Market-1501					CUHK03				
	rank-1	rank-5	rank-10	rank-20	mAP	rank-1	rank-5	rank-10	rank-20	mAP	rank-1	rank-5	rank-10	rank-20	mAP
supervised learning	63.4	78.9	83.3	87.4	42.6	76.2	89.5	93.3	95.8	53.1	24.8	41.1	52.4	64.2	23.2
Duke (baseline)	-	-	-	-	-	36.1	52.8	60.9	69.2	14.2	4.4	9.9	13.7	19.4	4.0
Duke (PUL)	-	-	-	-	-	44.7	59.1	65.6	71.7	20.1	numbers to be updated				
Market (baseline)	21.9	38.3	44.4	50.5	10.9	-	-	-	-	-	5.5	10.7	14.4	19.2	4.4
Market (PUL)	numbers to be updated					-	-	-	-	-	7.6	13.8	18.4	25.1	7.3
CUHK03 (baseline)	14.9	25.5	31.4	37.5	7.0	30.0	46.4	53.9	61.3	11.5	-	-	-	-	-
CUHK03 (PUL)	numbers to be updated					numbers to be updated					-	-	-	-	-

Table 1 Person re-ID accuracy (rank-1,5,10,20 precision and mAP) of three methods: the baseline (fine-tuned ResNet-50 on the same dataset for training/testing), PUL, and PPUL. Note that PUL and PPUL work on unsupervised settings which need the baseline model for initialization.



(a) (b) (c)

Fig. 5 The impact of λ on re-ID accuracy ($K = 750$). Since we use cosine distance, a larger λ means a stricter sample selection process, and vice versa. (a) Proportion of selected samples in the 1st training iteration; (b) proportion of selected samples during the whole training process; (c) performance changes during whole training process. The baseline is plotted in the dashed horizontal line.

# clusters (K)	250	500	750	1,000	1,250
rank-1	32.2	34.6	37.3	36.7	37.5
mAP	12.8	14.5	15.7	14.6	15.4

Table 2 Re-ID accuracy of PUL without sample selection ($\lambda = 0$, $K = 750$).

Results are presented in Table 2 and Fig. 5(c). Comparing with the baseline results in Table 1, we hardly observe any noticeable improvement of using PUL without sample selection. When $K = 1,250$, the largest gain over the baseline we can obtain is +1.4% in rank-1 accuracy, and +1.2% in mAP. This indicates that the samples within each cluster are very noisy, and that if we use all the noisy samples for fine-tuning without any selection, the learned CNN model does not show much improvement. These results suggest that reliable sample selection is a necessary component in PUL.

Further understanding of PUL with parameter changes. Two important parameters are involved in this paper, *i.e.,* the reliability threshold λ and the number of clusters K. To this end, we mainly deploy the fine-tuned ResNet-50 model on Duke described in Section 4.3. We apply PUL using this CNN model on Market-1501 without any labels. PUL is always trained for 25 iterations.

First, we evaluate the impact of K (we fix λ to 0.85), which is sampled from $\{250, 500, 750, 1,000, 1,250\}$, because Market-1501 has 751 training identities (not known in practice). Results are shown in Fig. 4. Since the Market-1501 dataset has 751 identities, $K = 750$ achieves the best performance as expected.

Second, we evaluate the impact of λ, keeping $K = 750$. Since we use the cosine distance to measure the similarity of two feature vectors and ResNet uses ReLU (rectified linear unit) as activation function, the range of λ should be $[0, 1]$. To further narrow the test range for λ, we present the proportions of selected samples under different λ in the first training iteration in Fig. 5(a). When $\lambda < 0.7$, nearly all of data instances are selected as being reliable, which is undesirable. Therefore, we choose λ from $\{0.70, 0.75, 0.80, 0.85, 0.90\}$. Fig. 5(b) demonstrates the change of the portion of selected samples during training. As training goes on, more and more samples are selected. Let us take $\lambda = 0.85$ for instance. Before the 9th training iteration, the proportion of selected samples increases fast from 31.8% to 45.3%. After that, the number oscillates between 45.3% and 47.4%. The curve becomes saturated after 20 epochs, which indicates that PUL

training can be stopped when the number of selected samples converges.

An interesting phenomenon in Fig 5(b) is that, when $\lambda \leq 0.85$, the proportion of reliable samples in the 1st training iteration witnesses a sudden drop after the 2nd training iteration. This is because, in the 1st training iteration, the original model fine-tuned on Duke is too weak to discriminate the data from Market. As a result, too many samples are incorrectly selected as reliable samples. In the 2nd training iteration, the model has been fine-tuned on Market and can better separate reliable and unreliable samples on Market.

We report the re-ID performance with different λ in Fig 5(c). In general, performance improves with more iterations, indicating that the model gradually gets stronger. Specifically, $\lambda = 0.85$ yields superior accuracy.

5 Conclusions

In this paper, a simple progressive unsupervised learning (PUL) method is proposed for person re-ID, based on the iterations between k-means clustering and CNN fine-tuning. We show that reliable sample selection is a key component to its success. Progressively, the proposed method produces CNN models with high discriminative ability.

The proposed method is easy to implement and a number of further extensions can be made. For example, as mentioned in [37], in video re-ID, frames within a tracklet can be viewed as belonging to the same ID, which can be used to initialize clustering and fine-tuning. Another promising idea is to integrate diversity into PUL, so that training samples from multiple cameras can be selected for fine-tuning.

References

1. Arthur, D., Vassilvitskii, S.: k-means++: the advantages of careful seeding. In: SODA (2007) 5
2. Bai, S., Bai, X., Tian, Q.: Scalable person re-identification on supervised smoothed manifold. In: CVPR (2017) 5
3. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: ICML (2009) 3
4. Chen, W., Chen, X., Zhang, J., Huang, K.: Beyond triplet loss: a deep quadruplet network for person re-identification. In: CVPR (2017) 1, 2
5. Cheng, D., Gong, Y., Zhou, S., Wang, J., Zheng, N.: Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In: CVPR (2016) 1
6. Fan, M., Deyu, M., Qi, X., Li, Z., Dong, X.: Self-paced cotraining. In: ICML (2017) 3
7. Felzenszwalb, P.F., Girshick, R.B., McAllester, D.A., Ramanan, D.: Object detection with discriminatively trained part-based models. TPAMI **32**(9) (2010) 5
8. Geng, M., Wang, Y., Xiang, T., Tian, Y.: Deep transfer learning for person re-identification. arXiv preprint arXiv:1611.05244 (2016) 1
9. Gray, D., Brennan, S., Tao, H.: Evaluating appearance models for recognition, reacquisition, and tracking. In: Proc. IEEE International Workshop on Performance Evaluation for Tracking and Surveillance (PETS) (2007) 1
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 3, 5
11. Hermans, A., Beyer, L., Leibe, B.: In defense of the triplet loss for person re-identification. CoRR (2017) 1, 2
12. Jiang, L., Meng, D., Yu, S., Lan, Z., Shan, S., Hauptmann, A.G.: Self-paced learning with diversity. In: NIPS (2014) 3
13. Kodirov, E., Xiang, T., Fu, Z., Gong, S.: Person re-identification by unsupervised l_1 graph learning. In: ECCV (2016) 1, 2
14. Kumar, M.P., Packer, B., Koller, D.: Self-paced learning for latent variable models. In: NIPS (2010) 2, 3
15. Li, W., Zhao, R., Xiao, T., Wang, X.: Deepreid: Deep filter pairing neural network for person re-identification. In: CVPR (2014) 4
16. Lin, Y., Zheng, L., Zheng, Z., Wu, Y., Yang, Y.: Improving person re-identification by attribute and identity learning. CoRR (2017) 1, 2
17. Liu, C., Change Loy, C., Gong, S., Wang, G.: Pop: Person re-identification post-rank optimisation. In: ICCV (2013) 2
18. Liu, H., Feng, J., Qi, M., Jiang, J., Yan, S.: End-to-end comparative attention networks for person re-identification. CoRR (2016) 1, 2
19. Ma, X., Zhu, X., Gong, S., Xie, X., Hu, J., Lam, K.M., Zhong, Y.: Person re-identification by unsupervised video matching. Pattern Recognition (2017) 1, 3
20. Peng, P., Xiang, T., Wang, Y., Pontil, M., Gong, S., Huang, T., Tian, Y.: Unsupervised cross-dataset transfer learning for person re-identification. In: CVPR (2016) 1, 3
21. Radenovic, F., Tolias, G., Chum, O.: CNN image retrieval learns from bow: Unsupervised fine-tuning with hard examples. In: ECCV (2016) 3
22. Ristani, E., Solera, F., Zou, R.S., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: ECCV (2016) 5
23. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: CVPR (2015) 3
24. Sun, Y., Zheng, L., Deng, W., Wang, S.: Svdnet for pedestrian retrieval. arXiv preprint arXiv:1703.05693 (2017) 1, 5
25. Varior, R.R., Haloi, M., Wang, G.: Gated siamese convolutional neural network architecture for human re-identification. In: ECCV (2016) 1, 2
26. Wang, H., Gong, S., Zhu, X., Xiang, T.: Human-in-the-loop person re-identification. In: ECCV (2016) 2
27. Wang, H., Zhu, X., Xiang, T., Gong, S.: Towards unsupervised open-set person re-identification. In: ICIP (2016) 3
28. Wang, T., Gong, S., Zhu, X., Wang, S.: Person re-identification by video ranking. In: ECCV (2014) 1
29. Xiao, T., Li, H., Ouyang, W., Wang, X.: Learning deep feature representations with domain guided dropout for person re-identification. In: CVPR (2016) 1, 2
30. Xiao, T., Li, S., Wang, B., Lin, L., Wang, X.: Joint detection and identification feature learning for person search. arXiv preprint arXiv:1604.01850 (2017) 2
31. Yang, Y., Li, S., Wen, L., Lyu, S.: Unsupervised learning of multi-level descriptors for person re-identification. In: AAAI (2017) 1, 2
32. Ye, M., Liang, C., Yu, Y., Wang, Z., Leng, Q., Xiao, C., Chen, J., Hu, R.: Person reidentification via ranking aggregation of similarity pulling and dissimilarity pushing. IEEE Transactions on Multimedia **18**(12), 2553–2566 (2016) 5
33. Zheng, L., Bie, Z., Sun, Y., Wang, J., Su, C., Wang, S., Tian, Q.: MARS: A video benchmark for large-scale person re-identification. In: ECCV (2016) 2, 4
34. Zheng, L., Huang, Y., Lu, H., Yang, Y.: Pose invariant embedding for deep person re-identification. arXiv preprint arXiv:1701.07732 (2017) 2
35. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: A benchmark. In: ICCV (2015) 1, 4

36. Zheng, L., Wang, S., Tian, L., He, F., Liu, Z., Tian, Q.: Query-adaptive late fusion for image search and person re-identification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1741–1750 (2015) 5
37. Zheng, L., Yang, Y., Hauptmann, A.G.: Person re-identification: Past, present and future. CoRR (2016) 1, 2, 5, 7
38. Zheng, Z., Zheng, L., Yang, Y.: Unlabeled samples generated by GAN improve the person re-identification baseline in vitro. CoRR (2017) 1, 2, 4, 5
39. Zhong, Z., Zheng, L., Cao, D., Li, S.: Re-ranking person re-identification with k-reciprocal encoding. In: CVPR (2017) 5